(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 936 530 A1

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
18.08.1999 Bulletin 1999/33

(51) Int. Cl.^{6}: G06F 1/00

(21) Application number: 98710001.3

(22) Date of filing: 16.02.1998

(84) Designated Contracting States:
AT BE CH DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE
Designated Extension States:
AL LT LV MK RO SI

(71) Applicant:
Siemens Nixdorf
Informationssysteme AG
33106 Paderborn (DE)

(72) Inventor: Benson, Glen
81739 München (DE)

(74) Representative:
Epping, Wilhelm, Dr.-Ing. et al
Patentanwalt
Postfach 22 13 17
80503 München (DE)

(54) Virtual smart card

(57) Smart card technology is in the process of emerging as a fundamental advance in computer security. A Virtual Smart Card emulates a real smart card by providing an identical interface and services. However, a Virtual Smart Card has no physical manifestation any smart card-aware application can seamlessly interoperate with either a real smart card or a Virtual Smart Card. A Virtual Smart Card Sever or a duplication-protected physical media communicates with the Virtual Smart Card in order to activate or to deactivate the Virtual Smart Card.

Fig 15

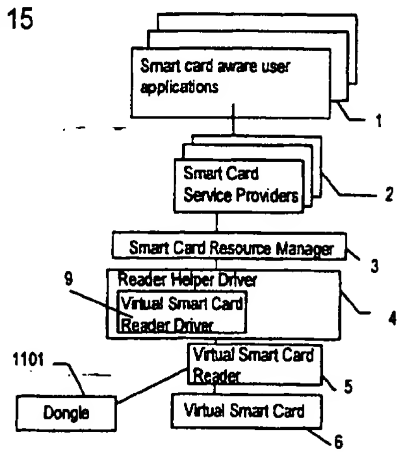
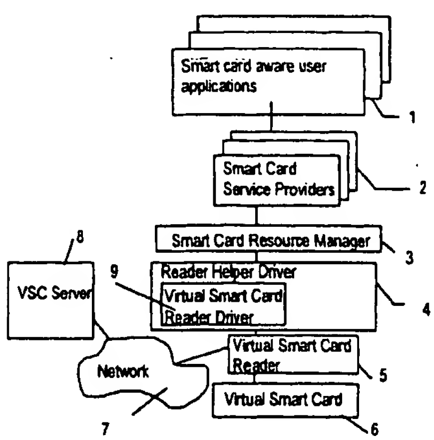


Fig 1



EP 0 936 530 A1

replace the destroyed smart card with an identical copy, or invalidate the lost smart card and issue a complete replacement.

[0008] One promising application of Smart Card technology is license and copy protection (LCP) as described in EP97710011.4. When the owner inserts his or her smart card, copy protected programs execute; and when the owner removes the smart card, the copy protected programs stop. So, the smart card acts as a "digital ignition key" that serves an analogous purpose to the ignition key in an automobile. A second promising application of Smart Card technology is Internet authentication. The owner authenticates him or herself to a remote machine by proving that he or she has the required smart card.

[0009] A problem with smart card technology is its inherent expense and logistic overhead. One cannot use a smart card until one physically attaches a computer to a smart card reader.

[0010] This problem is been solved by the features of claim 1 and claim 10.

[0011] The invention presents a bridge technology called Virtual Smart Card which emulates a real smart card by providing an identical interface and collection of services. However, a Virtual Smart Card has no physical manifestation. Any smart card-aware application can seamlessly inter-operate with either a real smart card or a Virtual Smart Card.

[0012] Although a Virtual Smart Card has no physical manifestation, a Virtual Smart Card emulates all three of the real smart card's states. An owner can insert a Virtual Smart Card with the effect that the Virtual Smart Card's state changes from idle to in-use. An owner can remove a Virtual Smart Card to change the state back from in-use to idle. After removing the Virtual Smart Card from one machine, the owner can potentially insert the Virtual Smart Card into a different machine. The owner cannot insert the Virtual Smart Card in the second machine until the owner removes the Virtual Smart Card from the first machine. If the owner's machine crashes, the owner may potentially lose his or her Virtual Smart Card. In this case, the owner usually can recover the lost Virtual Smart Card. However, in some rare cases, the Virtual Smart Card disappears and the owner must report the loss to the Virtual Smart Card issuing authority. The issuing authority responds in accordance to its policy, i.e., replacing the lost Virtual Smart Card with either an exact duplicate or a complete replacement.

[0013] The issuing authority operates a central trusted server called a Virtual Smart Card Server (VSC Server). The VSC Server maintains a database of all Virtual Smart Cards including the respective states and cryptographic keys. A Virtual Smart Card owner performs an insert operation by sending a request to the VSC Server for his or her Virtual Smart Card. The VSC Server mediates the request by first authenticating the owner and the Virtual Smart Card's implementation; and then validating that the requested Virtual Smart Card is currently in the idle state. If the authentication and mediation succeeds, then the VSC Server updates the database to indicate that the Virtual Smart Card is in-use. The VSC Server then permits the owner to use the Virtual Smart Card. When the Virtual Smart Card owner performs a remove operation, the Virtual Smart Card disables itself, securely sends a remove request to the VSC Server, and then shuts itself down. When the VSC Server receives a remove request, the VSC Server resets the Virtual Smart Card's state in the database to idle.

[0014] An alternative to the communication between the Virtual Smart Card and the Virtual Smart Card Server is presented in claim 10. The Virtual Smart Card Reader communicates with a Dongle (or some other duplication-protected physical media). A duplication protected physical media has the property that it is exceedingly difficult for an unauthorized attacker to construct a copy of the media. The Virtual Smart Card is a copy protected program that executes only if permitted by the Dongle. If the end-user attaches the Dongle to the machine, then the Virtual Smart Card executes; otherwise, the Virtual Smart Card stops.

[0015] A special extension to the claimed Virtual Smart Card is to augment the user authentication mechanism with a reader-less authentication device.

[0016] The advantage of this extension is excellent authentication at a low cost. The benefit is that the Virtual Smart Card architecture effectively extends the functionality of the reader-less device to include encryption.

[0017] As claimed one promising application of Virtual Smart Card technology is license and copy protection (LCP) . When the owner inserts his or her Virtual Smart Card, copy protected programs execute; and when the owner removes the Virtual Smart Card, the copy protected programs stop. So, the Virtual Smart Card acts as a "digital ignition key" that serves an analogous purpose to the ignition key in an automobile. A second promising application of Virtual Smart Card technology is Internet authentication. A common architecture exploited by many of today's enterprises is a corporate Intranet connected to the Internet via a firewall. In this architecture, an Intranet-located VSC Server distributes Virtual Smart Cards to machines physically located behind the corporate firewall. Once an owner inserts his or her Virtual Smart Card, the owner can exploit the Virtual Smart Card's cryptographic services to securely connect to Internet servers, Extranets, or generate digital signatures.

[0018] For a more complete understanding of the present invention and for further advantages thereof, reference is now made to the following Description of the Preferred Embodiments taken in conjunction with the accompanying Drawings in which:

FIG. 1 is a block diagram of the present system architecture of the virtual smart card sys-

a Virtual Smart Card 6, the VSC Server 8 downloads the protected information; and when the owner removes a Virtual Smart Card 6, the Virtual Smart Card 6 uploads the updated protected information to the VSC Server 8.

Encrypted Memory

[0027] Immediately after performing the insert operation, the Virtual Smart Card 6 generates a new, temporary symmetric key. Next, the Virtual Smart Card 6 decrypts the protected information using the protection key and re-encrypts the information using the temporary key. When performing the remove operation, if an update is required, the Virtual Smart Card decrypts the protected information using the temporary key and then re-encrypts the information using the protection key. The Virtual Smart Card 6 uploads the re-encrypted information to the VSC Server 8.

[0028] During the relatively short periods in which the Virtual Smart Card 6 needs the protected information, the Virtual Smart Card 6 decrypts the information using the temporary key. Next, the Virtual Smart Card 6 performs processing as required. If the processing modifies the protected information, then the Virtual Smart Card re-encrypts the information using the temporary key. Finally, the Virtual Smart Card 6 zeros out the plain text image. The Virtual Smart Card 6 repeats this procedure each time that it uses the protected information.

Volatile Memory

[0029] The Virtual Smart Card 6 stores its encrypted protected information in volatile memory (not shown) of the data processing unit or machine, e.g. a personal computer, where it runs. Before the Virtual Smart Card 6 exits, it explicitly zeros out all of its volatile memory used to store the protected information.

Wired Memory

[0030] A Virtual Smart Card 6 wires the memory that stores protected information. The wire operation precludes the memory from being paged out to swap space, e.g. at a hard disk of a personal computer.

Polling

[0031] A Virtual Smart Card 6 periodically polls its machine to ensure that an attacker has not copied the Virtual Smart Card 6 to a different machine. The Virtual Smart Card 6 stores a Machine Unique Key (MUK) in volatile memory. Periodically, the Virtual Smart Card 6 obtains a new MUK from the machine. If the new MUK does not match the old MUK, then the Virtual Smart Card 6 detects an attempted attack and exits. The MUK is a hash of information that uniquely identifies the machine, e.g., network address, machine name, number of sectors on each fixed disk, and size of swap space.

[0032] The Virtual Smart Card 6 performs a similar procedure using its Process ID. If the Virtual Smart Card 6 notices during polling that the queried Process ID does not match the stored Process ID, then the Virtual Smart Card 6 immediately exits.

[0033] The Virtual Smart Card periodically polls its host machine for the time of day. The Virtual Smart Card 6 compares its expectation with the result of the polling. If the result does not reasonably match expectations, then the Virtual Smart Card 6 shuts itself down by executing the remove operation. For example, if the Virtual Smart Card 6 polls the machine approximately every hour, then the Virtual Smart Card 6 would detect an error if the elapsed time between polling exceeds ninety minutes.

Virtual Smart Card (VSC) Server

[0034] The VSC Server 8 is a trusted application which maintains a database. It has to supervise one or more Virtual Smart Cards 6. In order to make the supervision possible each Virtual Smart Card 6 has the following records:

- Serial Number: The serial number is a unique identifier of a Virtual Smart Card 6.
- State: The state variable stores exactly one of the following values: in-use, idle, and destroyed. If state has the value in-use, then the VSC Server 8 recognizes that a Virtual Smart Card 6 has been inserted but not yet removed. The idle state indicates that the Virtual Smart Card 6 has been removed. The destroyed state indicates that the Virtual Smart Card 6 is no longer valid. The state of a destroyed Virtual Smart Card 6 never changes.
- MUK: The MUK is a machine unique key. If the state is either idle or destroyed, then the MUK gets the NULL value. If the state is in-use, then MUK value gets the MUK of the currently executing machine.
- Protected Information: The protected information contains information that the Virtual Smart Card 6 protects against attack. Examples of protected information are confidential encryption keys, or the state of electronic counters. The owner's protection key encrypts the protected information. Normally, the VSC Server 8 operators do not have access to an owner's protection key.
- Protected Channel Info: If the state of a the Virtual Smart Card 6 is idle or destroyed, then the protected channel info gets the NULL value. If the state is in-use, then the Protected Channel Info gets the

Keyfile Authentication

[0044] The Virtual Smart Card 6 obtains an license and copying protection (LCP)-compliant keyfile which contains the Virtual Smart Card's public key and a confidential authentication string as1, e.g., a 128-bit random number. The keyfile is signed using the VSC Server's private key and is encrypted using a proprietary symmetric algorithm. Systems using a keyfile are known from the Patent application EP97710011.4.

[0045] A software vendor locates the customer's public keying material and embeds the customer's public keying material into a keyfile and sends the keyfile to the customer by electronic mail. Once the customer installs the keyfile, the protection mechanism permits the customer to execute the protected software (provided that the customer can prove that he or she has access to the customer's private keying material via a probabilistic proof). The creation of the keyfile is performed by a keyfile generator, which is a program that executes at the vendor's facility. The vendor must take care to guard this program.

[0046] The Virtual Smart Card 6 decrypts the keyfile and validates the signature. Next, the Virtual Smart Card 6 decrypts and discovers the plaintext key as1. Next, the VSC Server 8 and the Virtual Smart Card 6 repeat the protocol described above in conjunction with Figure 2 with one exception. The VSC Server 8 and the Virtual Smart Card 6 substitute the confidential authentication string as1 for the master key.

[0047] The advantage of keyfile authentication is that the attacker does not compromise all Virtual Smart Cards by breaking the security of a single keyfile.

[0048] Any implementation of a Virtual Smart Card 6 should authenticate itself using both forms of authentication described above if a high grade of security assurance is wanted. However, in order to provide the best security assurance, one should additionally implement the advanced form of authentication listed below.

One-Time Algorithm

[0049] The VSC Server 8 authenticates a Virtual Smart Card 6 with the aid of mobile agents and automated code generation. A mobile agent is an executable code segment that passes between different machines, e.g., an Active X control. Automated code generation is vehicle by which one can generate a new executable at run-time. The automated code generator produces Virtual Smart Cards 6 according to a template which ensures that all Virtual Smart Cards 6 are identical in all but two respects:

> Authentication: Each Virtual Smart Card 6 has a unique authentication function, *f*. This function accepts a randomly generated number as input and produces a number as output. This output is suitable for deriving a key used in a symmetric encryption algorithm, e.g., DES.

> Wrapper: The portion of the Virtual Smart Card 6 executable that implements *f* is encrypted (with a hardcoded symmetric key): Immediately before executing *f*, the Virtual Smart Card 6 executable locates the hardcoded key and performs the required decryption. The Virtual Smart Card 6 zeros out the plaintext implementation of *f* immediately after execution.

[0050] At runtime, the VSC Server 8 generates two random numbers, *x* and *y*. The VSC Server 8 computes the following result:

$$w = E[x, f(y)],$$

where *E* is a symmetric encryption function, e.g., DES, *x* is a plaintext value, and *f(y)* is a value used to derive an encryption key. In other words, the VSC Server 8 computes *w* by encrypting *x* using the result of the computation *f(y)*. The VSC Server 8 passes *w* and *y* to the Virtual Smart Card 6 . Authentication succeeds only if the Virtual Smart Card 6 can discover *x* using the following decryption step within a short time period, e.g., 30 seconds:

$$x = D[w, f(y)].$$

That is, the Virtual Smart Card 6 decrypts w using a key derived from the result of the computation *f(y)*.

[0051] No two Virtual Smart Cards share the same function, *f*. Furthermore, no two installs of the same Virtual Smart Card 6 share the same function *f*. Each implementation of *f* should vary in terms of both operations and parameters. Furthermore, each implementation of *f* should be rather imposing from the perspective of a reverse engineering attack. An example specification of *f* is provided below:

$$f(y) = \frac{28734y}{\int_{23}^{87} \cos(y)^{19} dy}$$

[0052] An imposing function, *f*, would frustrate an attacker. The VSC Server 8 can quickly generate the required implementation given the aid of a good mathematics tool which automatically generates "C" implementations. For example, we specified the example function, *f*, using "Mathcad", Hompage ref:http://www.mathsoft.com/mathcad/ - a commercially available tool. Using this tool were able to generate ten

keyfile.

[0065] Some example applications of Software License and Copy protection system LCP using Virtual Smart Cards 6 are listed below.

- Try-Before-Buy: Before purchasing an application, a potential customer obtains a Try-Before-Buy demo. The keyfile for the demo permits limited usage in terms of either functionality or permitted executable period. Hopefully, if the potential customer likes the demo, then the customer subsequently purchases the software. The software vendor connects a VSC Server 8 to the Internet or another network that allows access to many computers. Anyone can connect to the Internet whenever he or she wishes, register with the software vendor, and obtain a unique Virtual Smart Card 6. The vendor downloads a corresponding keyfile to permit the end-user to execute the program.

- Network PCs and Network Computers: In order to decrease the total cost of ownership of computers, enterprises are beginning to administer machines using a client/server architecture. Each client regularly obtains programs and maintenance services from the central server. The server bears the responsibility of ensuring that each client runs correctly. One can add VSC Server 8 functionality to the central server without any significant increase in overhead because the architecture requires a network connection anyway.

- Licensed Software Repository: An enterprise's central server stores a collection of copy protected programs. Employees download the programs from the central server onto their machines. If the employee wishes to execute a program, then the employee purchases a keyfile. The enterprise distributes private keys to employees guarded by Virtual Smart Cards 6.

Network Authentication

Intranet VSC Server

[0066] The potential applications of Virtual Smart Cards 6 are not limited to software copy protection. Figure 3 illustrates an enterprise that operates a VSC Server 8 in an intranet 11 behind a firewall 10 which protects the intranet 11 from the internet 12. The purpose of the architecture is to deploy asymmetric cryptography throughout the enterprise without bearing the cost of smart cards. Some employees have a Personal Computer PC. Each employee can use the services of his or her Virtual Smart Card 6 to authenticated to remote nodes, communicate via secured electronic mail, electronically sign documents, or use copy protected programs.

[0067] The security of the architecture significantly exceeds the security employed by most enterprises today because of the reduced dependence upon passwords. For most purposes, in lieu of authenticating using a password, an enterprise employe can authenticate using his or her Virtual Smart Card's 6 private key.

[0068] The weakest point in the architecture is the employee's authentication to the VSC Server 8. Ultimately, the employee must supply his or her password. Nevertheless, one can optionally configure a VSC Server 8 to require additional authentication material, e.g., a properly registered MUK, or coordination with an external authentication method such as the Secure ID system.

Internet Service Provider VSC Server 8

[0069] An Internet Service Provider (ISP) is an ideal candidate for operating a VSC Server 8. When one of the ISP's customers connects to the ISP, the customer automatically inserts a Virtual Smart Card 6; and at disconnect time, the Virtual Smart Card 6 automatically removes itself. The customer may subsequently execute copy protected programs, securely access network services, and participate in electronic commerce.

[0070] The ISP can provide the VSC Server 8 using minimal extensions to its existing customer database. With the exception of a little extra processing at customer login time, the Virtual Smart Card 6 service requires no ISP resources.

Telephony over an Internet Protocol network(H.323)

[0071] The upcoming trend in enterprise telephony is to replace traditional telephone technology e.g., PBXs with a telephone to LAN gateway. Enterprise employees connect their telephones, picture phones, and computers PC to their intranet 11; and a gateway connects the intranet to external networks such as the telephone network, the Internet 12, and Asynchronous Transfer Modus (ATM) networks.

[0072] A gateway and gatekeeper standard that provides telephony over an Internet Protocol (IP) network is the *H.323*. *H.323* defines a gateway that translates IP traffic to and from the telephone network; and *H.323* defines a gatekeeper that mediates and helps route traffic through the gateway. Unfortunately, by connecting the intranet to both the telephone network and the Internet, the enterprise unwittingly creates one of the most valuable resources potentially available to a hacker. Consider, for example, a hacker who breaks through the enterprise's Internet firewall 10 and accesses the intranet 11. This hacker may potentially place telephone calls from this intranet 11 thereby building a telephone gateway to the world.

[0073] Virtual Smart Card 6 technology can provide an important countermeasure to such an attack. By implementing support for asymmetric authentication in

a success code and continues processing the insert operation 104.

User Authentication (501 Figure 8)

[0085] Multiple mechanisms exist for authorizing a user (smart card owner). One such mechanism is illustrated in Figure 9. The smart card owner enters a password 601 (a confidential string). The Virtual Smart Card 6 program extracts a confidential value called SALT 602 from its own executable. All Virtual Smart Card 6 programs have the same SALT 602. The Virtual Smart Card 6 program computes 603 the MD5 hash algorithm over the password 601 and the SALT 602. The result is a 128 bit value. The Virtual Smart Card 6 program extracts the first 64 bits and names these bits the authentication key 604. The Virtual Smart Card 6 program extracts the second 64 bits and names these bits the protection key 605.

[0086] The Virtual Smart Card authenticates the smart card owner by proving to the VSC Server that the Virtual Smart Card knows the authentication key. The simplest such "proof" is to simply send the authentication key to the VSC Server. Since the communication channel is protected 301, one need not be concerned with an intruder who listens for passwords. The VSC Server simply validates the authentication key against its internal table. More complex password authentication schemes also exist [Menezes, A., Oorschot, P., and Vanstone, S., Handbook of Applied Cryptography, CRC Press, Boca Raton 1996]

Implementation Authentication (503 Figure 8)

[0087] Multiple mechanisms exist for authenticating the Virtual Smart Card's implementation. One such mechanism is illustrated in Figure 10. The VSC Server 8 generates a new, unique random number ri1 701. The VSC Server 8 sends ri1 701 to the Virtual Smart Card 6. The Virtual Smart Card 6 has a confidential Master key 702 embedded within the Virtual Smart Card 6 executable image. All Virtual Smart Cards 6 have the same Master Key 702 embedded within their own executable. The Virtual Smart Card 6 generates a new, unique random number ri2 704. The Virtual Smart Card 6 computes the hash, e.g., MD5, of ri1 701, ri2 704, and Master 702. The Virtual Smart Card 6 returns ri2 704 and the result of the hash to the VSC Server. The VSC Server 8 recomputes the hash using ri1 701 and ri2 704. If the recomputed hash matches the value returned by the Virtual Smart Card 6 then the authentication step succeeds. This authentication step proves to the VSC Server that the Virtual Smart Cards knows the Master Key. An attacker cannot build a rogue implementation of a Virtual Smart Card without first disassembling a Virtual Smart Card and discovering the Master Key.

Machine Unique Key MUK (303 Figure 6):

[0088] The Virtual Smart Card 6 computes a machine unique key (MUK) 303 of its local machine. First the Virtual Smart Card 6 extracts the following values from its machine: the network address, the machine's name, the currently logged in user (if applicable), and the number of sectors on each fixed drive. The MUK 303 is the hash of all of the extracted information.

Machine Lock (304 Figure 6):

[0089] The Virtual Smart Card 6 opens a well-known path for exclusive access in the local machine's registry. The registry is a resource available in Windows 95™ or Windows NT™ with separately identified items. The same well-known path is hardcoded into every implementation of a Virtual Smart Card 6 program. Only one Virtual Smart Card 6 program at a time can open the path for exclusive access.

[0090] One may implement a Virtual Smart Card 6 to execute on a machine other than Windows 95™ or Windows NT™. In this case, in lieu of the registry, the Virtual Smart Card 6 obtains exclusive access to some other well-known resource, e.g., a file. The idea is that the well-known resource helps cooperating Virtual Smart Cards 6 ensure that only one Virtual Smart Card 6 executes on a machine at a time.

Mediation (305 Figure 6):

[0091] The VSC Server 8 looks up the Virtual Smart Card 6 in the VSC Server's 8 database using the Serial Number provided during Authentication 302. If the state of the Virtual Smart Card 6 is not idle then the VSC Server 8 refuses the request and returns a negative acknowledgment. The Virtual Smart Card 6 then exits.

[0092] If the state of the Virtual Smart Card 6is idle, then mediation succeeds and the VSC Server 8 proceeds to setting the state of the Virtual Smart Card 6 to in-use 307.

Set state to in-use (307 Figure 6):

[0093] The VSC Server 8 updates the record in the VSC Server's database for the Virtual Smart Card 6 by setting the state to in-use.

Enable and download (308 Figure 6):

[0094] The VSC Server 8 returns a positive acknowledgment and the Virtual Smart Card 6 begins servicing its owner. The VSC Server 8 also downloads information that is protected using the protection key 605. The Virtual Smart Card 6 uses the protection key 605 to decrypt. Subsequently, the Virtual Smart Card 6 can access the protected information.

image of the protected information 1011. Otherwise, if the processing updated the protected information, then the Virtual Smart Card 6 encrypts 1012 the new version of the protected information using temp. The Virtual Smart Card 6 overwrites the old version of the protected information.

[0101] In the following some modifications are described

Copying

[0102] The administrator of the VSC Server 8 can potentially make multiple copies of a single Virtual Smart Card 6. The administrator simply builds entries in its database for new Virtual Smart Cards 6 but copies the same information in each entry.

Implementation not requiring a VSC Server

[0103] In Figure 1, the Virtual Smart Card Reader 5 communicates with the VSC Server via the Network. However, one may potentially change the architecture such that the Virtual Smart Card Reader 5 does not communicate with the VSC Server 8 via the Network. Instead, the Smart Card Service Provider 2, the Smart Card Resource Manager 3, the Reader Helper Driver 4, or the Virtual Smart Card Reader Driver 9 could potentially communicate via the VSC Server 8 via the network while providing the same network services as described in the embodiment of the invention.

[0104] Figure 15 illustrates an alternative implementation of the Virtual Smart Card 6. This implementation does not require a VSC Server 8.

[0105] Instead of communicating with the Virtual Smart Card Server 8 the Virtual Smart Card Reader 5 communicates with duplication-protected physical media, e.g., a Dongle 1101. A duplication protected physical media 1101 has the property that it is exceedingly difficult for an unauthorized attacker to construct a copy of the media 1101. The Virtual Smart Card 6 is a copy protected program that executes only if permitted by the Dongle 1101. If the end-user attaches the Dongle 1101 to the machine, then the Virtual Smart Card 6 executes; otherwise, the Virtual Smart Card 6 stops.

[0106] The states and state transitions of the Virtual Smart Card 6 are illustrated and described in relation to Figure 4.

Idle 101: The Virtual Smart Card 6 does not execute. The Virtual Smart Card 6 cannot validate the Dongle 1101. Possibly, the Dongle 1101 is not currently installed.

In-Use 102: The Virtual Smart Card 6 is executing. The Virtual Smart Card 6 periodically communicates with the attached Dongle 1101 as illustrated and described in relation to Figure 14.

Destroyed 103: The Dongle 1101 that authorizes a machine's Virtual Smart Card 6 has been destroyed or lost.

[0107] The operations of the Virtual Smart Card 6 are described below:

Insert 104: The end-user attaches the Dongle 1101 and boots the Virtual Smart Card 6 program. The Virtual Smart Card 6 program does not operate unless the Virtual Smart Card 6 program can validate that the Dongle 1101 is present. The state of the Virtual Smart Card 6 is in-use 102 after the Virtual Smart Card 6 detects the Dongle 1101. This state is not explicitly recorded as in the case with the VSC Server 8.

Remove 105: The Dongle 1101 fails to authorize the Virtual Smart Card 6. For example, the end-user either removes the Dongle 1101, or the Virtual Smart Card 6 shuts down. The state is idle 101 after the Dongle 1101 is removed.

Recover 106: If the end-user loses his or her Dongle 1101, then the end-user can request a replacement from the Dongle 1101 issuing authority. Presumably, the authority that first placed the encryption key on the Dongle 1101 remembers the Dongle's key. The state is idle 101, once the end-user obtains a new Dongle 1101.

Destroy 107: The Dongle 1101 is lost or physically destroyed. The state is Destroyed 103 after the Dongle 1101 is physically lost or destroyed.

[0108] When the Virtual Smart Card 6 is idle 101, the Dongle 1101 stores the protected information. The Dongle 1101 has two storage locations. The first storage location stores the Dongle 1101 key (see Figure 14) and the second storage location has the protected information.

[0109] The protected information is encrypted using a symmetric encryption key called VSC-Key. When the Virtual Smart Card 6 boots, the Virtual Smart Card 6 executes the insert operation. Upon successful completion of the insert operation, the Virtual Smart Card 6 enters the in-use 102 state.

[0110] When the Virtual Smart Card 6 is in the in-use 102 state, the Virtual Smart Card 6 obtains protected information. Normally, the Virtual Smart Card 6 stores this protected information on the Dongle 1101 in encrypted form. When the Virtual Smart Card 6 wishes to obtain the protected information, the Virtual Smart Card 6 retrieves the protected information from the Dongle 1101. The Virtual Smart Card 6 uses a Master key to decrypt the protected information. The Master key is hardcoded into the Virtual Smart Card's 6 executable image. The Virtual Smart Card 6 stores its protected

Fig 1

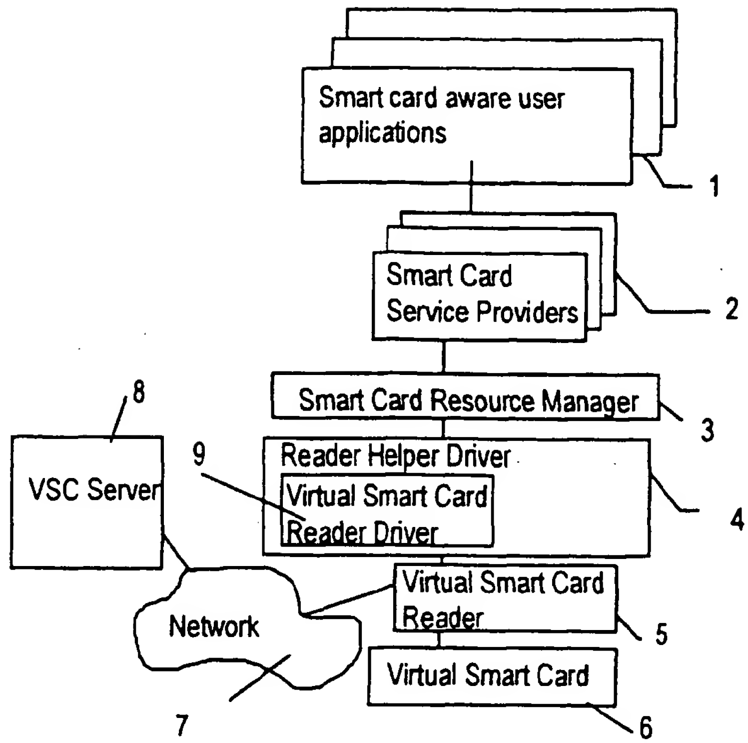

Smart card aware user applications
1
Smart Card Service Providers
2
Smart Card Resource Manager
3
8
VSC Server
9
Reader Helper Driver
Virtual Smart Card Reader Driver
4
Network
Virtual Smart Card Reader
5
Virtual Smart Card
7
6

Fig 2

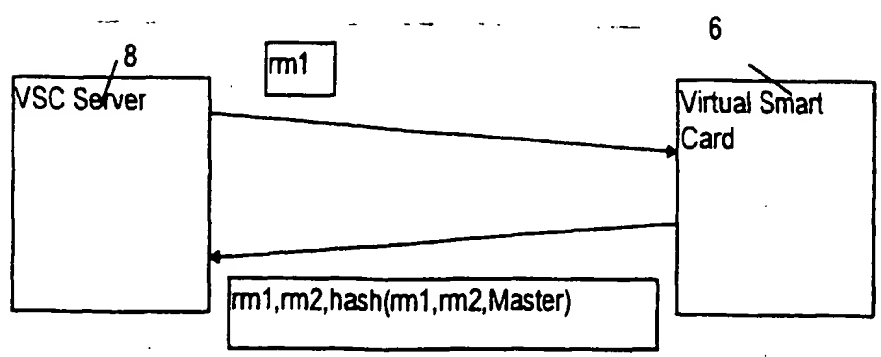

6
8
rm1
VSC Server
Virtual Smart Card
rm1,rm2,hash(rm1,rm2,Master)

Fig 5

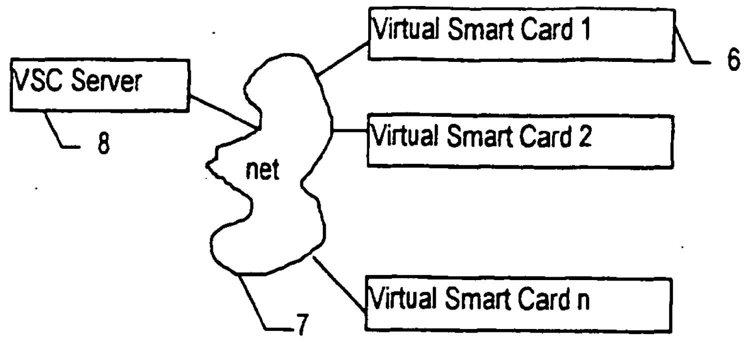

VSC Server
8
net
7
Virtual Smart Card 1
6
Virtual Smart Card 2
Virtual Smart Card n

Fig 6

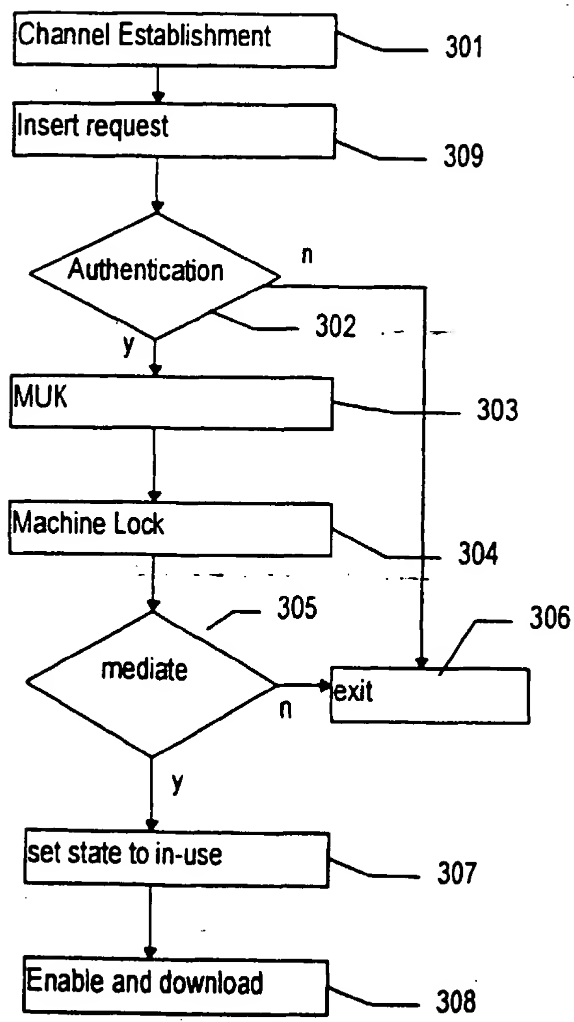

Channel Establishment
301
Insert request
309
Authentication
n
302
y
MUK
303
Machine Lock
304
305
306
mediate
exit
n
y
set state to in-use
307
Enable and download
308

Fig 9

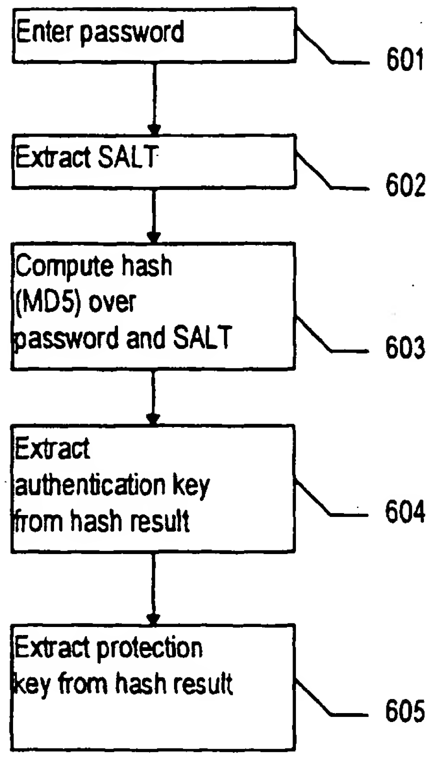

Enter password
601
Extract SALT
602
Compute hash
(MD5) over
password and SALT
603
Extract
authentication key
from hash result
604
Extract protection
key from hash result
605

Fig 10

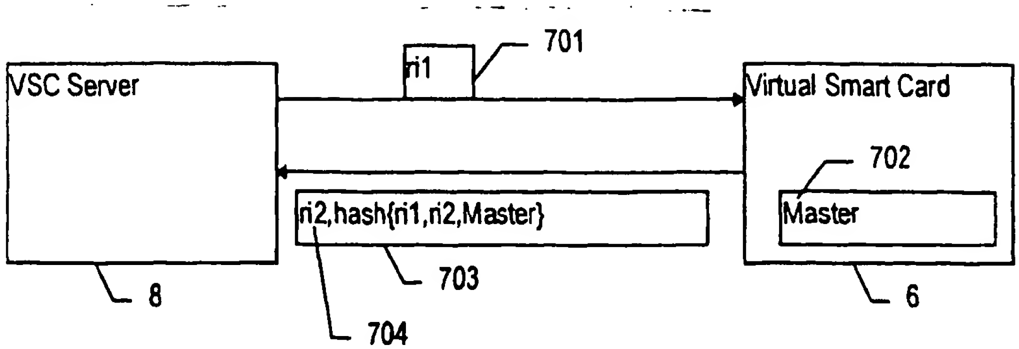

VSC Server
ri1
701
Virtual Smart Card
702
Master
ri2,hash{ri1,ri2,Master}
703
704
8
6

Fig 12

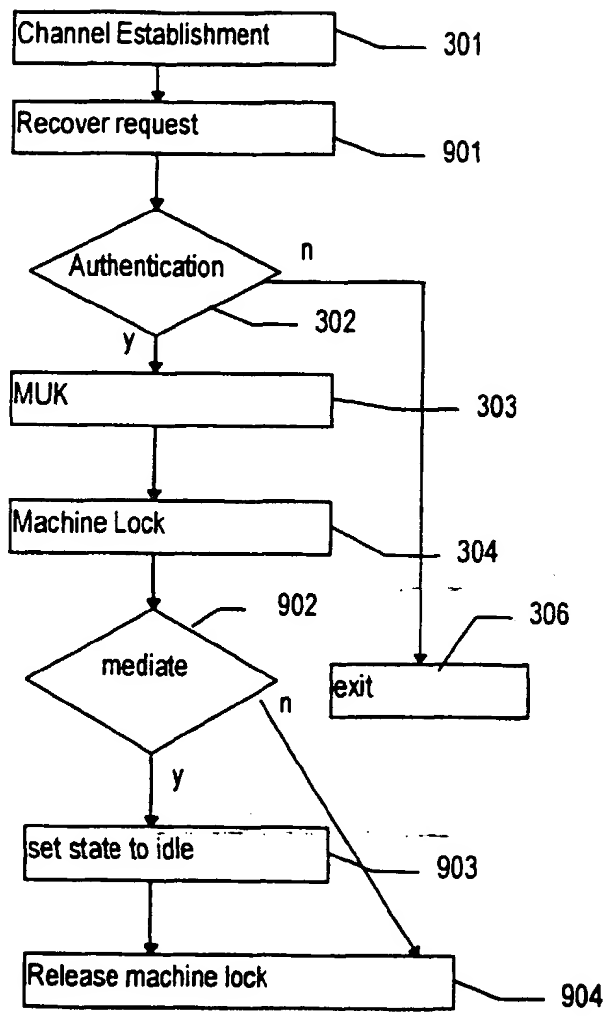
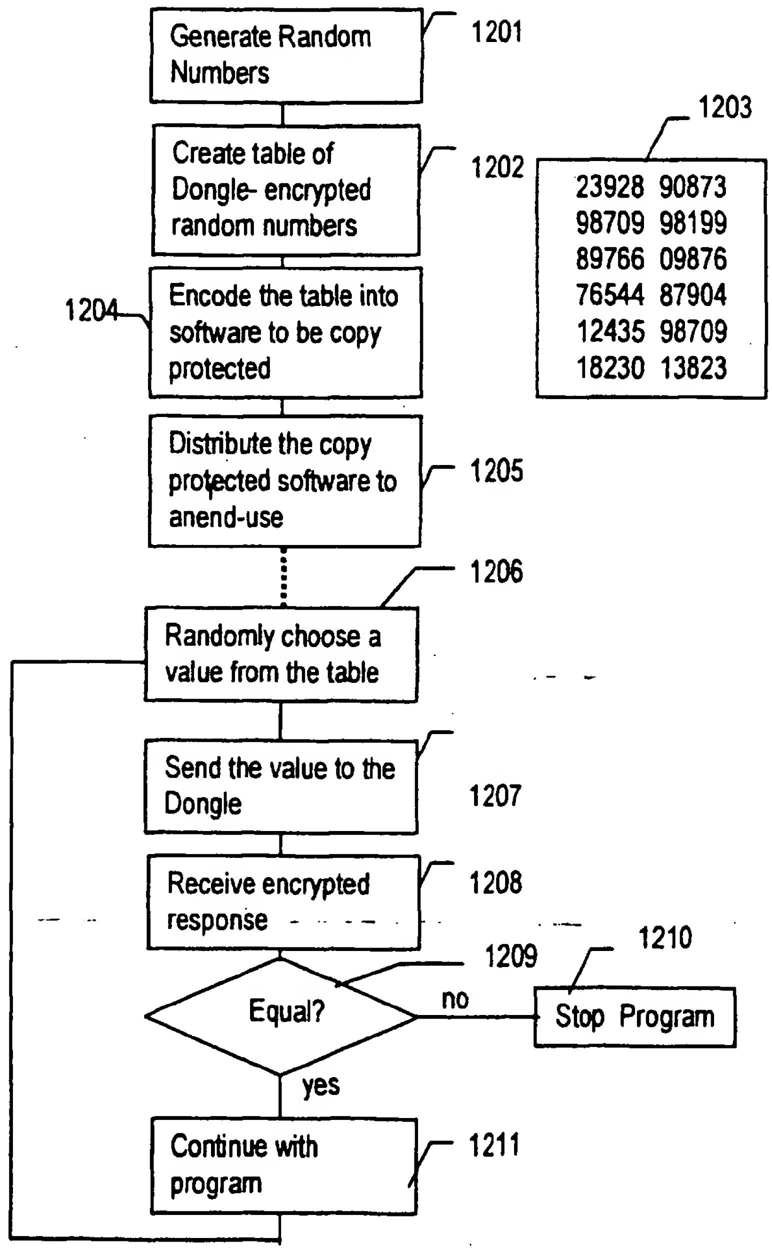

Channel Establishment
301
Recover request
901
Authentication
n
302
y
MUK
303
Machine Lock
304
902
306
mediate
exit
n
y
set state to idle
903
Release machine lock
904

Fig 14


Generate Random Numbers
1201
Create table of Dongle- encrypted random numbers
1202
1203
23928 90873
98709 98199
89766 09876
76544 87904
12435 98709
18230 13823
1204
Encode the table into software to be copy protected
Distribute the copy protected software to anend-use
1205
1206
Randomly choose a value from the table
Send the value to the Dongle
1207
Receive encrypted response
1208
1209
1210
Equal?
no
Stop Program
yes
Continue with program
1211

European Patent Office

EUROPEAN SEARCH REPORT

Application Number
EP 98 71 0001

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
A	WO 97 21162 A (NORTHERN TELECOM LTD) 12 June 1997 * the whole document *	1-10	G06F1/00
A	EP 0 752 635 A (SUN MICROSYSTEMS INC) 8 January 1997 * the whole document *	1-10	
A	WO 96 41445 A (SPYRUS INC) 19 December 1996 * the whole document *	1-10	
A	US 5 590 199 A (KRAJEWSKI JR MARJAN ET AL) 31 December 1996 * abstract; figure 5 * * column 3, line 41 - column 4, line 42 *	1-10	
A	US 5 191 611 A (LANG GERALD S) 2 March 1993 * the whole document *	1-10	
A	WO 95 12169 A (VISA INT SERVICE ASS) 4 May 1995		TECHNICAL FIELDS SEARCHED (Int.Cl.6) G06F
A	EP 0 561 685 A (FUJITSU LTD) 22 September 1993		

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
THE HAGUE	5 November 1998	Powell, D

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons

& : member of the same patent family, corresponding document